Scout: A Hardware-Accelerated Language for Data Visualization and Analysis

Patrick M^{c}Cormick[1], Jeff Inman[1], James Ahrens[1], JoAnn Lysne[1], Mathew Hecht[1], Mat Maltrud[1]
Tony Mezzacappa[2], John Blondin[3], Chuck Hansen[4], Greg Roth[4]

[1] Los Alamos National Laboratory, [2] Oak Ridge National Laboratory,
[3] North Carolina State University, [4] University of Utah.

Summary

The primary goal of the Scout Project is to provide efficient and effective techniques for the visualization and analysis of scientific data. Many visualization techniques rely on the use of indirect mappings to transform data into the final rendered imagery. In many situations it is preferable to express these mappings as mathematical expressions that can be directly applied to the data. Scout provides a data parallel language that provides such capabilities and exploits the computational power of graphics hardware to perform portions of the tasks that would otherwise be executed on the central processing unit.

The analysis of scientific data involves the investigation of the relationships between the numerical and spatial properties of one or more data sets. The use of visualization techniques also plays a key role in this process. Many different visualization methods use indirect mappings to assign optical properties (like color and transparency) to data values. While these techniques can be powerful, they have often had limited acceptance in the scientific community because they require scientists to work in a secondary, visualization-specific, data space that makes it difficult to efficiently express queries and mathematical operations.

We have designed and developed Scout, a software environment and programming language that provides expression-based queries that are evaluated directly in the original data space. In Scout a query consists of a set of relational and conditional expressions based on numerical values. The language reduces the computational bottleneck by utilizing the graphics processor (GPU) not only for rendering, but also as a computational co-processor responsible for offloading portions of the work that would traditionally be executed on the CPU. The language-based interface allows scientists to process multivariate data, express derived data, and define the associated mappings to the final imagery. This is all done in a language environment that is more familiar than many graphical user interfaces. In addition to reducing the workload on the CPU, the performance of Scout also benefits from the computational rates of the GPU. In our initial experiments, the graphics card has routinely outperformed the CPU. This performance improvement allows scientists to interactively explore their data using both qualitative and quantitative techniques.

Multivariate Visualization

When analyzing computational model results, it is often valuable for scientists to study the relationship between several different variables. This can be especially true when exploring anomalies and temporal data. A good example of such a situation is the data produced by the POP ocean model that simulates the occurrence of an El Niño-Southern Oscillation (ENSO) event.

ENSO events are well known for having an impact on the weather patterns of the Earth. Figure 1, on the top of the following page, shows the results of using Scout to study the differences between an ENSO event and the average ocean conditions produced by the POP model. The resulting image was produced using a Scout program that is only 15 lines long.

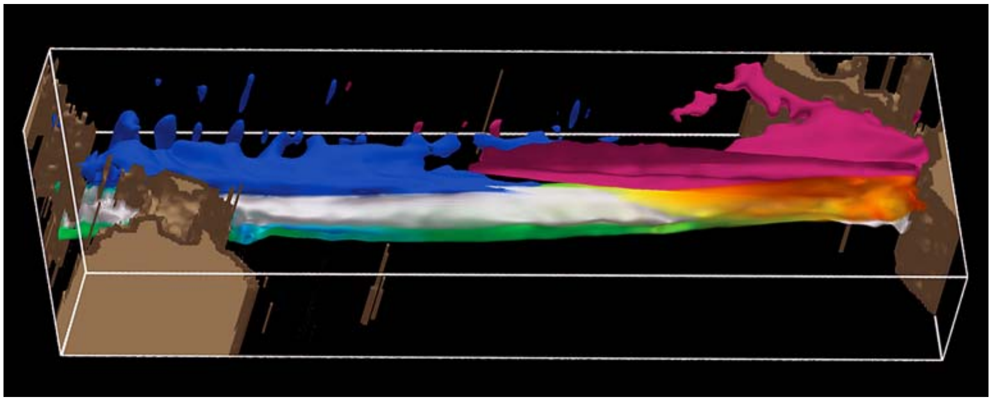

Figure 1. *The results produced by using Scout to study several features that are consistent with an El Niño Event. Anomalies in water temperature are shown as solid magenta and blue and are clipped south of the equator. The white region represents the ENSO thermocline and the color mapped region shows the mean January thermocline; the colors show the difference in sea surface heights. The thermocline is a sharp temperature gradient separating the upper layers of ocean water from those at depth. Land masses are rendered in tan.*

Computation of Derived Data

In the process of analyzing data it is often necessary for scientists to compute one or more derived values. These computations are often done before the actual analysis process and in many cases, are not integrated into visualization and analysis tools. When using Scout, it is possible to incorporate these steps directly into the workflow, thus allowing for a more flexible and efficient process.

The Terascale Supernova Initiative (TSI) project is currently studying the mechanics responsible for driving core collapse supernova. One important derived field is the entropy of a star's core as it explodes. Figure 2 presents an image produced by Scout of the derived entropy data. A simple Scout program, consisting of only a few lines of code, computed the entropy at a rate 3.5 times faster than a highly optimized version of the same calculations running on a 3.0 GHz Pentium Xeon EM64T processor.

Summary

Within the Scout Project, we are leveraging our broad expertise in the visualization and analysis of scientific data. This multidisciplinary approach is key to establishing a solid understanding of the overall challenges. In addition, by taking advantage of the power of today's commodity graphics cards and providing a flexible language-based interface we have built the foundation for high-performance, efficient, cost effective visualization and analysis.

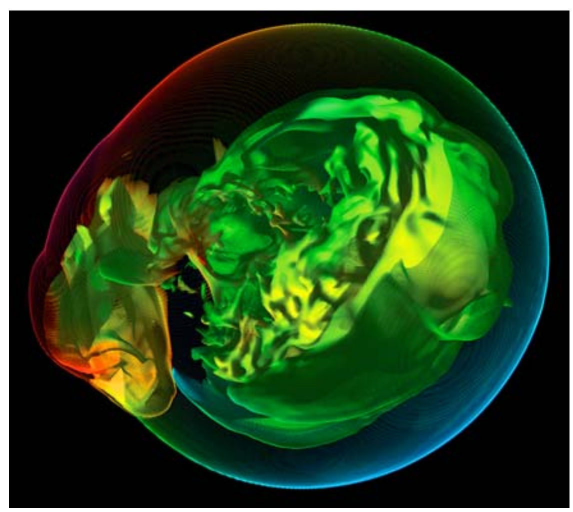

Figure 2. *The results of using Scout to compute entropy and render selected regions of interest colored by velocity magnitude.*

For further information on this subject contact:
Patrick McCormick, Principal Investigator
Los Alamos National Laboratory
Phone: 505-665-0201
pat@lanl.gov